



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/843,036	04/25/2001	Matthew Frank Trapani	00-625-B	7211

7590 01/28/2008
George I. Lee
McDonnell Boehnen Hulbert & Berghoff
32nd Floor
300 S. Wacker Drive
Chicago, IL 60606

EXAMINER

TRAN, QUOC A

ART UNIT	PAPER NUMBER
----------	--------------

2176

MAIL DATE	DELIVERY MODE
-----------	---------------

01/28/2008

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 09/843,036	Applicant(s) TRAPANI ET AL.	
	Examiner Tran A. Quoc	Art Unit 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 12 November 2007.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-19, 23-35 and 37-47 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 1-15 and 41-43 is/are allowed.
- 6) ☒ Claim(s) 16-19, 23-35, 37-40 and 45-47 is/are rejected.
- 7) ☒ Claim(s) 44 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 12 November 2007 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

This action is a **Non-Final** rejection in response to Amendment/Remarks filed on 11/12/2007.

In Applicant's Responses dated 11/12/2007, Applicant has amended independent claims 1, 16, added new claims 44-47, cancelled claims 20-22, 36, and argued against all rejections previously set forth in the Office Action dated 05/16/2007.

Claims 1, 16, 23, 27, 31, and 39 are independent claims; filing date 04-25-2001, priority date 04-26-2000 (Assignee Novara).

Specification

The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter.

Claim 44, recites the limitation "***the most recent document***" (see Lines 14-15 Pages 11). However, there is not any support for the phrase "***the most recent document***" in the Applicant's disclosure. See 37 CFR 1.75(d) (1) and MPEP § 608.01(o). Correction is required.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-19 and 41-44 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 1-15 and 41-44:

The language of the claims raise a question as to whether the claims are directed merely to an abstract idea that would not result in a practical application producing a concrete, useful, and tangible result to form the basis of statutory subject matter under 35 U.S.C. 101.

In summary, Claim 1 recites a "*system*" for executing instructions that perform various functions. The examiner notes that, although the "*instructions*" that the recited system executes are "*stored on a computer-readable medium*," the recited system itself is not tangibly embodied on a computer-readable medium (see Claim 1, Lines 1-3). Thus, for purposes of examination, the examiner interprets the recited "*system*" to be software per se. That is, the recited "*system*" is not a process, a machine, a manufacture or a composition of matter.

Accordingly, Claim 1 fails to recite statutory subject matter as defined in 35 U.S.C. 101.

Claims 2-15 and 41-44 merely recite further functions performed by the "*instructions*" and further describe the electronic information processed by the "*instructions*." Thus, Claims 2-15 and 41-44 do not further define the recited

"*system*" as being within a statutory process, machine, manufacture or composition of matter.

Accordingly, Claims 2-15 and 41-44 fail to recite statutory subject matter as defined in 35 U.S.C. 101.

Claims 16-19:

The language of the claims raise a question as to whether the claims are directed merely to an abstract idea that would not result in a practical application producing a concrete, useful, and tangible result to form the basis of statutory subject matter under 35 U.S.C. 101.

In summary, Claim 16 recites a "*system*" for executing instructions that perform various functions. The examiner notes that, although the "*instructions*" that the recited system executes are "*stored on a computer-readable medium*," the recited system itself is not tangibly embodied on a computer-readable medium (see Claim 16, Lines 1-3). Thus, for purposes of examination, the examiner interprets the recited "*system*" to be software per se. That is, the recited "*system*" is not a process, a machine, a manufacture or a composition of matter.

Accordingly, Claim 16 fails to recite statutory subject matter as defined in 35 U.S.C. 101.

Claims 17-19 merely recite further functions performed by the "*instructions*" and further describe the electronic information processed by the

"instructions." Thus, Claims 17-19 do not further define the recited "system" as being within a statutory process, machine, manufacture or composition of matter.

Accordingly, Claims 17-19 fail to recite statutory subject matter as defined in 35 U.S.C. 101.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 16-19, 23-35, 37-40, and 45-47 are rejected under 35 U.S.C. 103(a) as being unpatentable by **Bickmore** et al. US006857102B1-filed 01-29-1999 (hereinafter "Bickmore"), in view of **Hirsch** US 6,836,768 Provisional No. 60/186,052-filed 02-29-2000 (hereinafter "Hirsch").

Regarding ***independent claim 16***, Bickmore teaches:

An automatic normalizer for applying pattern recognition

(See Bickmore at Column 4 Line 65→ Column 5, discloses Automatic document re-authoring.)

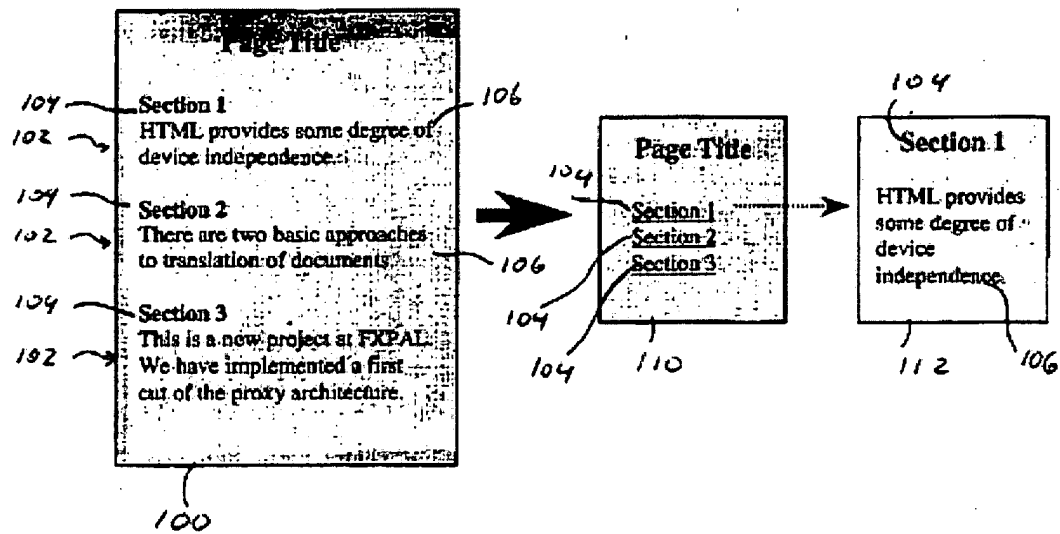


FIG. 1

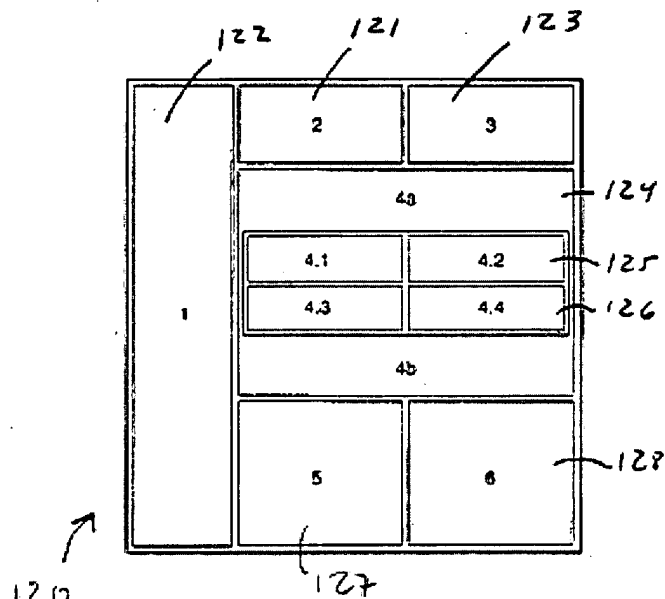


FIG. 2

**and weighting heuristics on the document tree to produce a
normalized document tree,**

(See Bickmore at Fig. 18 and Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document.

**and wherein the document tree is organized into a set of
hierarchical nodes having respective weights, where a weight
determines whether a node will be inserted into the normalized
document tree as a folder title or folder contents,**

(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations (i.e. AST) to achieve the best-looking document for a given display size.

Also see Bickmore at Column 8, Line 15→Column 9, Line 20, illustrates re-authoring of a document into a section list page and a number of section pages according to one exemplary embodiment of the document re-authoring

systems and methods of this invention; wherein a layout table that can be re-authored into a plurality of linked cells according to one exemplary embodiment of the document re-authoring systems and methods of this invention;)

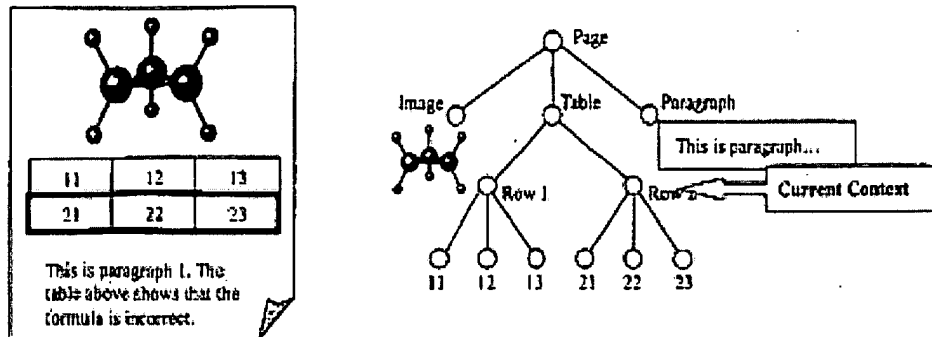


FIG. 17

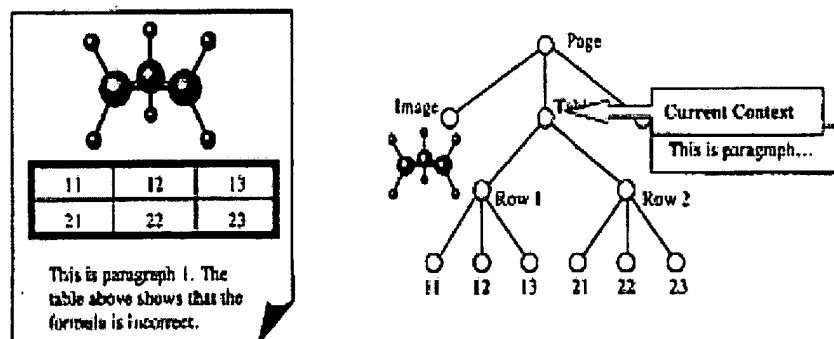


FIG. 18

wherein the document tree represents a format for displaying information content in a document on a device,
(See Bickmore at Column 4, Lines 30-45, discloses a set of structural page transformations (i.e. AST) to achieve the best-looking document for a given display size.)

and wherein the pattern recognition and weighting heuristics alter the format for displaying the information content,
(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations (i.e. AST) to achieve the best-looking document for a given display size. using the broadest reasonable interpretation, the Examiner equates the claimed **normalization markup** as equivalent to Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document as taught by Bickmore.)

In addition, Bickmore does not explicitly teach, but Hirsch teaches:

and wherein the automatic normalizer folderizes the information content by assigning content having a higher visibility of display a weight indicative of a folder title, wherein if a node has no effect on a visual display of the information content and the node is not folder contents, the node is removed.
(See Hirsch at Column 5, Lines 35-50, discloses a template on a very small display device (such as a cell phone) may only display one note at a time or a present a very few choices, while a viewer on a larger display may see many

notes. In various embodiments, templates may be used by a server-side process to influence the format and content downloaded to a particular client or by client-side processes to arrange and present received notes data.

Also see Hirsh at Column 13, Lines 30-50, discloses automatic note extraction rules may be constructed with a weighting system to evaluate structure, as is known in the art. Weights can be assigned to weighted rules based on any technique known for weighting; rules may be built on structural identifiers (i.e. folder title or content).

See also Hirsh at Column 7, Lines 35-65, discloses different views composed of different categories are dynamically presented to users based on user requests or on the content of the page. For different structured sources, categories can change as well. For example, on a news story, there may be several categories such as: source, key facts, key numbers, etc., while on a small page there may be only three categories of surfnodes: source, short text (page preview) and page links.

Using the broadest reasonable interpretation the Examiner equates the **claimed if a node has no effect on a visual display of the information content and the node is not folder contents, the node is removed as equivalent to surfnodes as taught by Hirsh.**

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching of Bickmore, to include a means of applying the automatic normalizer folderizes the information content by assigning content having a higher visibility of display a weight

indicative of a folder title, wherein if a node has no effect on a visual display of the information content and the node is not folder contents, the node is removed as taught by Hirsh. One of ordinary skill in the art would have been motivated to perform such a modification to provides an automatic re-authoring of web documents to make them automatically converts web-based documents designed for desktop viewing into formats appropriate for handheld devices with small display screens, such as Palm-PCs, PDAs, and cellular phones (as taught by Bickmore at page 534).

*Regarding **independent claim 23**, Bickmore teaches:*

**using a processor executing one or more instructions for:
determining if the information content contains normalization
markup, and if so: utilizing normalization markup in the information
content to normalize the information content, wherein the
normalization markup provide at least one specific instruction for
normalizing the information content.,**

(See Bickmore at Column 4 Line 65→ Column 5, discloses Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document, designed to be displayed on a desktop-sized monitor, along with characteristics of the target display device, and re-author the arbitrary document through a series of transformations, so that the arbitrary document can be appropriately displayed on the target display device.

Also see Bickmore at Fig. 1-2 and Column 8 lines 40-50, disclosed Indexed Segment transform, as much style information as possible is retained for the output elements, by outputting each element embedded within all of its ancestor partitions' HTML tags. The Indexed Segment transform then constructs an index page by copying a section header or first sentence from each element to be output, concatenating the copied portion onto an index page, and creating a hypertext link from each copied portion to the appropriate sub-page. Using the broadest reasonable interpretation, the Examiner equates the claimed **normalization markup** as equivalent to Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document as taught by Bickmore.

In addition, Bickmore does not explicitly teach, but Hirsch teaches:

**matching and applying a template to the information content,
wherein the template defines modifications to the document in
order to adapt the document for display on a device other than an
originally intended device, and if unsuccessful:**

(See Hirsch at Column 5, Lines 35-50, discloses a template on a very small display device (such as a cell phone) may only display one note at a time or a present a very few choices, while a viewer on a larger display may see many notes. In various embodiments, templates may be used by a server-side process to influence the format and content downloaded to a particular client or by client-side processes to arrange and present received notes data.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching of Bickmore, to include a means of matching and applying a template to the information content, wherein the template defines modifications to the document in order to adapt the document for display on a device other than an originally intended device as taught by Bickmore. One of ordinary skill in the art would have been motivated to perform such a modification to provides an automatic re-authoring of web documents to make them automatically converts web-based documents designed for desktop viewing into formats appropriate for handheld devices with small display screens, such as Palm-PCs, PDAs, and cellular phones (as taught by Bickmore at page 534).

Regarding independent claim 27,

the rejection of claim 16 is fully incorporated.

In addition, Bickmore teaches:

determining parent-child relationships between the weighted nodes based on the weighted nodes to produce a normalized document tree, wherein a weighted node is established as a child of a parent having the lightest weight of all the parents that is also greater than the weight of the weighted node.

(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set

of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document by Bickmore at page 534).

Regarding independent claim 39,

the rejection of claim 23, is fully incorporated.

In addition, Bickmore teaches:

**Applying changes to the document tree according to the
template markup language.**

(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation,

which may be in a document format that is different from the input format of the original document by Bickmore at page 534).

Claim 17, Bickmore teaches:

**the automatic normalizer comprises a markup assisted
normalizer for processing normalization markup in the document
tree to produce the normalized document tree,**

(See Bickmore at Column 2, Line 65 → Column 3, Line 5 discloses Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document, designed to be displayed on a desktop-sized monitor, along with characteristics of the target display device, and re-author the arbitrary document through a series of transformations, so that the arbitrary document can be appropriately displayed on the target display device.)

Claim 18, Bickmore teaches:

**the normalization markup does not affect a page for display by
a PC-based browser that utilizes hypertext markup language (HTML).**

(See Bickmore at Column 2, Line 65 → Column 3, Line 5 discloses Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document, designed to be displayed on a desktop-sized monitor, along with characteristics of the target display device, and re-author the arbitrary document through a series of transformations, so

that the arbitrary document can be appropriately displayed on the target display device.)

Claim 19, Bickmore teaches:

the normalized document tree represents a hierarchical representation of information in the document tree.

(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document by Bickmore at page 534).

Claims 24-26,

the rejection of claims 16, and 39 are fully incorporated, and are similarly rejected along the same rationale.

Claims 28-30,

the rejection of claims 16, 31 and 39 are fully incorporated, and are

similarly rejected along the same rationale.

In addition: Bickmore teaches:

weighting nodes in a table; and attempting to match the table to a predefined pattern of weights, and if successful: extracting data in response to the predefined pattern.

(See Bickmore at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document.)

Claim 40, Bickmore teaches:

the document tree is represented by a plurality of nodes, and wherein applying changes to the document tree according to the template markup language comprises adding a node to the plurality of nodes, dropping at least one of the plurality of nodes, moving at least one of the plurality of nodes, partitioning at least one of the plurality of nodes into folders, or calling user defined formatting rules on at least one of the plurality of nodes.

(See Bickmore at Column 4 Line 65 → Column 5, discloses Automatic document re-authoring.

Also See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document.

Also see Bickmore at Column 2, Line 65 → Column 3, Line 5 discloses Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document, designed to be displayed on a desktop-sized monitor, along with characteristics of the target display device, and re-author the arbitrary document through a series of transformations, so that the arbitrary document can be appropriately displayed on the target display device.)

Claim 45, Bickmore teaches:

wherein the normalization markup indicates which sections of the information content are to be modified for display on the device other than the originally intended device.

(See Bickmore at Column 2, Line 65 → Column 3, Line 5 discloses Automatic document re-authoring involves developing software that can take an arbitrary document, such as an HTML document, designed to be displayed on a desktop-sized monitor, along with characteristics of the target display device, and re-author the arbitrary document through a series of transformations, so that the arbitrary document can be appropriately displayed on the target display device.

Also See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, then apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document.

Claim 46, Bickmore does not expressly teach, but Hirsh teaches:

**wherein the normalization markup includes meta-tags
embedded in the information content that provide at least one
specific instruction for normalizing the information content.**

(See Hirsch at Column 13, Line 65 discloses layout parsing is interpreting the markup language or encoding language in which the document is written (such

as HTML, XML (i.e. META TAGS), RTML, RTF, MS Word, etc.) to correctly identify information that is of interest for further parsing. This type of parsing must be done by all web browsers to correctly display page content, even those that do not attempt to further extract information on a page

Also see Hirsch at Column 5, Lines 35-50, discloses a template on a very small display device (such as a cell phone) may only display one note at a time or a present a very few choices, while a viewer on a larger display may see many notes. In various embodiments, templates may be used by a server-side process to influence the format and content downloaded to a particular client or by client-side processes to arrange and present received notes data.

Also see Hirsh at Column 13, Lines 30-50, discloses automatic note extraction rules may be constructed with a weighting system to evaluate structure, as is known in the art. Weights can be assigned to weighted rules based on any technique known for weighting; rules may be built on structural identifiers (i.e. folder title or content).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching of Bickmore, to include a means of applying the automatic normalizer folderizes the information content by normalization markup includes meta-tags embedded in the information content that provide at least one specific instruction for normalizing the information content as taught by Hirsh. One of ordinary skill in the art would have been motivated to perform such a modification to provides an automatic re-authoring of web documents to make them automatically converts web-based

documents designed for desktop viewing into formats appropriate for handheld devices with small display screens, such as Palm-PCs, PDAs, and cellular phones (as taught by Bickmore at page 534).

Claim 47, Bickmore does not expressly teach, but Hirsh teaches:

**wherein the meta-tags include instructions including
extracting data from a table, creating a folder with a given title, or
moving content to a marked position.**

(See Hirsch at Column 13, Line 65 discloses layout parsing is interpreting the markup language or encoding language in which the document is written (such as HTML, XML (i.e. META TAGS), RTML, RTF, MS Word, etc.) to correctly identify information that is of interest for further parsing. This type of parsing must be done by all web browsers to correctly display page content, even those that do not attempt to further extract information on a page

Also see Hirsch at Column 5, Lines 35-50, discloses a template on a very small display device (such as a cell phone) may only display one note at a time or a present a very few choices, while a viewer on a larger display may see many notes. In various embodiments, templates may be used by a server-side process to influence the format and content downloaded to a particular client or by client-side processes to arrange and present received notes data.

Also see Hirsh at Column 13, Lines 30-50, discloses automatic note extraction rules may be constructed with a weighting system to evaluate structure, as is known in the art. Weights can be assigned to weighted rules

based on any technique known for weighting; rules may be built on structural identifiers (i.e. folder title or content).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching of Bickmore, to include the meta-tags include instructions including extracting data from a table, creating a folder with a given title, or moving content to a marked position as taught by Hirsh. One of ordinary skill in the art would have been motivated to perform such a modification to provides an automatic re-authoring of web documents to make them automatically converts web-based documents designed for desktop viewing into formats appropriate for handheld devices with small display screens, such as Palm-PCs, PDAs, and cellular phones (as taught by Bickmore at page 534).

Claims 31-35, and 37 are rejected under 35 U.S.C. 103(a) as being unpatentable by Chen et al, U.S. Patent No US 6,507,856 filed 06/05/1999 (hereinafter, "Chen"), in view of Hanson et al. US 20050268224A1- Continuation of No. 09/481,840 filed 01/12/2000 (hereinafter "Hanson").

Regarding independent claim 31, Chen teaches:

using a processor executing one or more instructions for:
receiving data; and storing information relating to the data into an
array;

(See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array.)

**wherein the plurality of array utilizes re-usable buffers, and
wherein the stored information describes the document object tree
and tree dependencies as a mutable object,**

(See Chen at Column 3, Lines 35-67; discloses receiving data; and storing information relating to the data into an array, wherein automating document exchange and merging uses extensible Markup Language (XML) documents, XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a document merging algorithm.)

**and wherein the document object tree includes a hierarchical
set of nodes that define a format for displaying information content
in a document on a device,**

(See Chen at Column 4, Lines 1-10, discloses DOM tree or the array. Looping includes reusing a format for data entry)

**and wherein an array contain values associated with the nodes
of the data, and wherein operations on the nodes can be carried out
by utilizing the value as referenced to the affected nodes,**

(See Chen at Column 3, Lines 35-65, discloses DOM tree parsing or serialization prepares the first document in a suitable data structure, such as tree or array, for efficient processing and matching.)

wherein separate arrays are used to store values representing properties of each node including properties selected from the group consisting of a parent node, a previous sibling node, a next sibling node, and a first child node.

(See Chen at Column 3, Lines 35-65, discloses DOM tree parsing or serialization prepares (i.e. properties of nodes) the first document in a suitable data structure, such as tree or array, for efficient processing and matching.

Also see Chen at Column 4, Lines 10-20, discloses he DTD parser transforms the DTD with repeatable and optional fields into a template in tree structure or serialized array with special markers around loop header nodes or name tags. Optional fields may include a second business address or phone number.)

receiving data; and storing information relating to the data into an array, wherein an array utilize re-usable buffers, and wherein the stored information describes the document object tree and tree dependencies as a mutable object,

(See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array.

Also see Chen at Column 3, Lines 35-65, discloses DOM tree parsing or serialization prepares (i.e. properties of nodes) the first document in a suitable data structure, such as tree or array, for efficient processing and matching.

Also see Chen at Column 4, Lines 10-20, discloses he DTD parser transforms the DTD with repeatable and optional fields into a template in tree structure or serialized array with special markers around loop header nodes or name tags. Optional fields may include a second business address or phone number.)

In addition, Chen does not explicitly teach, but Hanson teaches:

plurality of arrays,

(See Hanson at Para 67, discloses the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Chen's DOM tree array, to include a means of formatting web document Object in HTML for transmitted HTML in the client/server environment utilizing the Object Framework provides a set of objects for developing Web documents, objects are provided for low

level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers- See Hanson at the Abstract and at Para 67.)

Claim 32, Chen teaches:

transforming the document object tree, wherein the transformed document object tree is represented by the single mutable object;

See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array, wherein automating document exchange and merging uses extensible Markup Language (XML) documents, XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a document merging algorithm.)

Claim 33, Chen teaches,

adding an array as to the DOM received data grows in size;

(See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array, wherein automating document exchange and merging uses extensible Markup Language (XML) documents,

XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a document merging algorithm.)

In addition, Chen does not explicitly teach, but Hanson teaches:

plurality of arrays,

(See Hanson at Para 67, discloses the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Chen's DOM tree array, to include a means of formatting web document Object in HTML for transmitted HTML in the client/server environment utilizing the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers- See Hanson at the Abstract and at Para 67.)

Claim 34, Chen teaches:

**wherein array is used to hold the associated values that
represent properties of the nodes of the document object tree;**

(See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array, wherein automating document exchange and merging uses extensible Markup Language (XML) documents, XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a document merging algorithm.)

In addition, Chen does not explicitly teach, but Hanson teaches:

plurality of arrays,

(See Hanson at Para 67, discloses the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Chen's DOM tree array, to include a means of formatting web document Object in HTML for transmitted

HTML in the client/server environment utilizing the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers- See Hanson at the Abstract and at Para 67.)

Claim 35, Chen teaches:

**referencing a re-usable content buffer that contains data;
wherein an array store start and end positions of data that reference
the data stored in the re-usable content buffer;**

(See Chen at Column 3, Lines 35-67, discloses receiving data; and storing information relating to the data into an array, wherein automating document exchange and merging uses extensible Markup Language (XML) documents, XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a document merging algorithm.)

In addition, Chen does not explicitly teach, but Hanson teaches:

plurality of arrays,

(See Hanson at Para 67, discloses the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for

traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle

errors and events, and user interface objects such as buttons and scrollers.)

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified Chen's DOM tree array, to include a means of formatting web document Object in HTML for transmitted HTML in the client/server environment utilizing the Object Framework provides a set of objects for developing Web documents, objects are provided for low level graphics and much higher level aspects of graphics, including objects for traditional data types such as integers, characters, strings, as well as arbitrary collections of OBJECTS, ARRAYS, list and vectors, objects that handle errors and events, and user interface objects such as buttons and scrollers- See Hanson at the Abstract and at Para 67.)

Claim 37, Chen teaches:

normalizing the document object tree model by a template

normalizer for applying templates to the document object tree;

(See Chen at Column 3, Lines 35-67, discloses automating document exchange and merging uses extensible Markup Language (XML) documents, XML name tag map table design, Document Object Model (DOM) tree parsing or serialization, return document template generation, constraint set design, and a

document merging algorithm. Also Chen Fig. 7 item 365 discloses the uses of template normalizing.)

Claim 38 are rejected under 35 U.S.C. 103(a) as being unpatentable by Chen et al, U.S. Patent No US 6,507,856 filed 06/05/1999 (hereinafter, "Chen"), in view of Hanson et al. US 20050268224A1- Continuation of No. 09/481,840 filed 01/12/2000 (hereinafter "Hanson"), further in view of Bickmore et al. US006857102B1-filed 01-29-1999 (hereinafter "Bickmore").

Claim 38,

Chen does not expressly teach, but Bickmore teaches:

normalizing the document object tree model by an automatic normalizer for applying pattern recognition and weighting heuristics on the document tree to produce a normalized document tree;

(See Bickmore at Column 4, Lines 30-45, discloses dynamically re-authors requested web pages using a heuristic planning technique and a set of structural page transformations to achieve the best-looking document for a given display size. Further Bickmore discloses at Column 11, Lines 60-65, the re-author document by first parsing the document and constructing a parse tree or abstract syntax tree (AST) representation of the document, the apply a series of transformations to the parse tree (i.e. a heuristic planning technique) and map each resulting transformed parse tree back into a document representation, which may be in a document format that is different from the input format of the original document.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the teaching of Chen and Hanson, to includes a means of normalizing the document object tree model by an automatic normalizer for applying pattern recognition and weighting heuristics on the document tree to produce a normalized document tree as taught by Bickmore. One of ordinary skill in the art would have been motivated to perform such a modification to provides an automatic re-authoring of web documents to make them automatically converts web-based documents designed for desktop viewing into formats appropriate for handheld devices with small display screens, such as Palm-PCs, PDAs, and cellular phones (as taught by Bickmore at page 534).

It is noted that any citations to specific, pages, columns, lines, or figures in the prior art references and any interpretation of the references should not be considered to be limiting in any way. A reference is relevant for all it contains and may be relied upon for all that it would have reasonably suggested to one having ordinary skill in the art. See, MPEP 2123.

Response to Arguments

Applicant's arguments with respect to claim(s) 16-19, 23-35, and 37-40 have been considered but are moot in view of the new ground(s) of rejection (See above rejection for details).

Allowable Subject Matter

Claims 1-15, and 41-44 include allowable subject matter and would be allowed if the 101 rejections for these claims are obviated.

Also, in regard to Claim 44, the objection to the Specification indicated in the above discussion must be obviated before this claim is allowed.

The following is a statement of reasons for the indication of allowable subject matter:

Under the broadest reasonable interpretation of the claimed limitation consistence with the Applicant's Specification, the prior art of record fail to teach all of the Applicant's claimed limitation (Claim(s) 1-15, and 41-43. In particularly, the claimed invention advantageously provides a finer level of detail that enables a desktop focus web content to handheld browsers, this requires filtering unsupported content, dropping unneeded content, reordering and partitioning content to improve navigation and application flow for allow the content source to be redefined once for all networks and device types display on a limited device. More particularly, a template normalizer utilizes regular expression pattern-matching to ***assign the content having a higher visibility a weight indicative of a folder title***, and If the similar nodes exist between the information content in the document and a previously normalized document, and if so, the automatic normalizer collapses the information content in the document in a manner similar to the previously normalized document (i.e. a template normalizer utilizes regular expression pattern-matching to impose a template over a document and attempts

to match the template to the document that is dynamically formats content to a form that is optimized for a particular electronic device that formatted to suit industry standard browsers, or targeted to an electronic device using the client side browser (See Claims 1, 16, and 27 and the Applicant Specification Page 35, Lines 4-15). (i.e. collapses information content utilizes the trees of the documents are compared to determine if similar fragments (list of links, table, image) exist. The similar fragments of the tree are collapsed into folders or select input elements. The effect is to conserve display space on the device.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Quoc A. Tran whose telephone number is 571-272-8664. The examiner can normally be reached on Monday through Friday from 9 AM to 5 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Doug Hutton can be reached on 571-272-4137. The fax phone number for the organization where this application or proceeding is assigned is (571)-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Quoc A, Tran/
Patent Examiner
Art Unit 2176
01/22/2008

/Doug Hutton/
Doug Hutton
Supervisory Primary Examiner
Technology Center 2100